Gowin_EMPU(GW1NS-4C) Hardware Design

Reference Manual

IPUG932-2.0E, 03/14/2024

Disclaimer

GOWINSEMI assumes no liability and provides no warranty (either expressed or implied) and is not responsible for any damage incurred to your hardware, software, data, or property resulting from usage of the materials or intellectual property except as outlined in the GOWINSEMI Terms and Conditions of Sale. GOWINSEMI may make changes to this document at any time without prior notice. Anyone relying on this documentation should contact GOWINSEMI for the current documentation and errata.

Revision History

Date	Version	Description
04/20/2020	1.0E	Initial version published.
02/08/2021	1.1E	● AHB PSRAM Memory Interface peripheral supported. ● AHB HyperRAM Memory Interface peripheral supported. ● APB SPI Nor Flash peripheral supported. ● GPIO supports multiple port types. ● I^2C supports multiple port types. ● ARM Keil MDK as well as GOWIN MCU Designer upgraded.
06/21/2021	1.2E	● Known issue of SPI full duplex read and write fixed. ● Synplify Pro removed. ● FPGA software upgraded. ● Reference design updated.
12/16/2022	1.3E	● Known issue of port signal name fixed. ● MCU IP updated and logic resources optimized. ● Software development kit updated.
03/14/2024	2.0E	● System clock frequency and performance boosted. ● Hardware reference design updated.

Contents

Contents i

List of Figures iii

List of Tables iv

1 Hardware Architecture 1

1.1 System Architecture 1

1.2 System Feature 2

1.2.1 MCU Core System 2

1.2.2 FPGA Core System 3

1.3 System Port 3

1.4 System Resource Utilization and Performance Statistics 8

2 Hardware Design Flow 9

2.1 Hardware Target 9

2.2 Software Version 9

2.3 IP Core Generator 9

2.4 Programmer 9

2.5 Design Flow 9

3 Project Template 11

3.1 Project Creation 11

3.1.1 Create a New Project 11

3.1.2 Set Project Name and Path 12

3.1.3 Select Device 12

3.1.4 Project Creation Completed 13

3.2 Hardware Design 14

3.2.1 TPIU Configuration 17

3.2.2 NVIC Configuration 17

3.2.3 GPIO Configuration 18
3.2.4 UART Configuration 19
3.2.5 AHB2 Master / Slave Configuration 20
3.2.6 SRAM Configuration 21
3.2.7 I2C Configuration 22
3.2.8 SPI Configuration 23
3.2.9 RTC Configuration 24
3.2.10 APB2 Master Configuration 24
3.2.11 PSRAM Memory Interface 26
3.2.12 HyperRAM Memory Interface 27
3.2.13 SPI-Flash Memory 28
3.3 User Design 29
3.4 Constraint 29
3.5 Configuration 29
3.5.1 Synthesize Configuration 29
3.5.2 Place Configuration 30
3.5.3 Route Configuration 31
3.5.4 Bitstream Configuration 32
3.6 Synthesize 33
3.7 Place & Route 33
3.8 Download 34
4 Reference Design 37

List of Figures

Figure 1-1 System Architecture 1
Figure 3-1 Create a FPGA Design Project 11
Figure 3-2 Set Project Name and Path 12
Figure 3-3 Select Device 13
Figure 3-4 Project Creation Completed 13
Figure 3-5 Select Gowin_EMPU(GW1NS-4C) IP Core 14
Figure 3-6 System Architecture 15
Figure 3-7 TPIU Configuration 17
Figure 3-8 NVIC Configuration 18
Figure 3-9 GPIO Configuration 19
Figure 3-10 UART Configuration 20
Figure 3-11 AHB2 Master/Slave Configuration 21
Figure 3-12 SRAM Configuration 22
Figure 3-13 I²C Configuration 23
Figure 3-14 SPI Configuration 23
Figure 3-15 RTC Configuration 24
Figure 3-16 APB2 Master[1-12] Configuration 25
Figure 3-17 Synthesize Configuration 30
Figure 3-18 Place Configuration 31
Figure 3-19 Route Configuration 32
Figure 3-20 Bitstream Configuration 32
Figure 3-21 Synthesize 33
Figure 3-22 Place & Route 34
Figure 3-23 Download Configuration for GW1NS-4C/GW1NSR-4C 35
Figure 3-24 Download Configuration for GW1NSER-4C 35

List of Tables

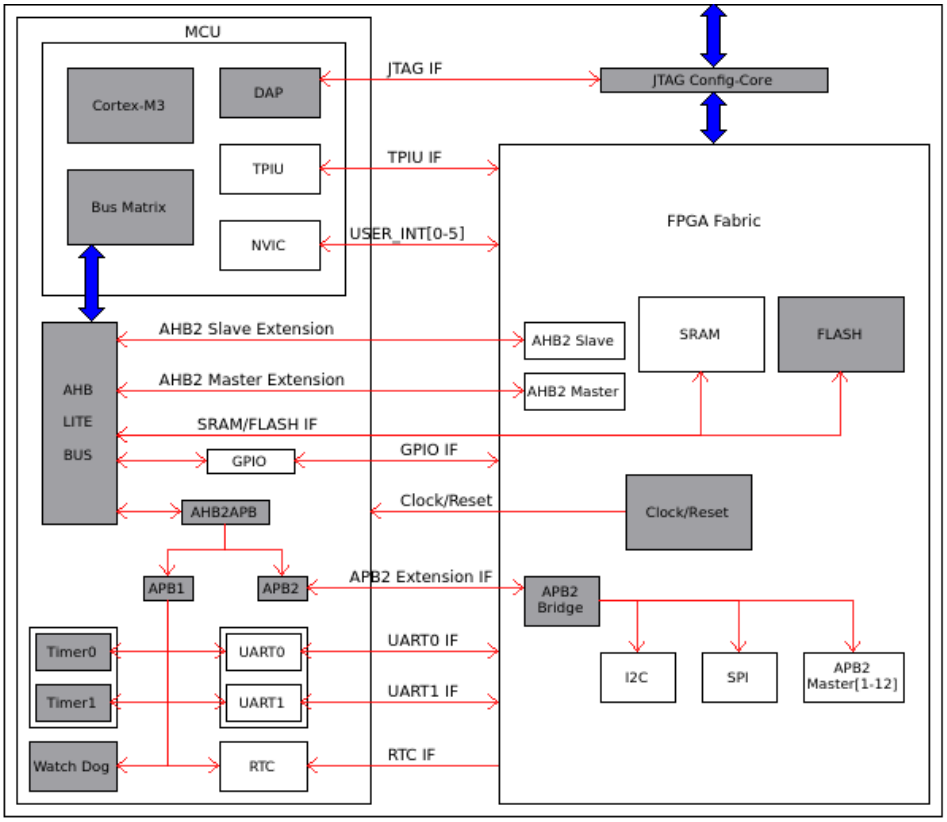
Table 1-1 Definition of System Ports 3
Table 1-2 System Resource Utilization and Performance Statistics 8
Table 3-1 System Configuration Options 16
Table 3-2 Base Address Mapping of APB2 Master [1-12] 26

1 Hardware Architecture

1.1 System Architecture

Gowin_EMPU(GW1NS-4C) is an on-chip system consisting of MCU core system and FPGA core system, as shown in Figure 1-1.

Figure 1-1 System Architecture



MCU core system consists of MCU Core, AHB and peripherals, AHB2APB Bridge, APB1 and peripherals.

FPGA core system consists of the clock and reset signal input, data

SRAM, instruction FLASH of MCU core system, APB2 Bridge, APB2 and peripherals.

1.2 System Feature

Gowin_EMPU(GW1NS-4C) includes two sub-systems:

- MCU core system
- FPGA core system

1.2.1 MCU Core System

MCU core system includes:

- MCU Core:
  - ARM Cortex-M3 Core, ARM architecture v7-M Thumb2 supporting 16-bit and 32-bit instruction set
  - DAP (Debug Access Port)
  - Bus Matrix
  - NVIC (Nested Vector Interrupt Controller)
  - TPIU (Trace Port Interface Unit)
- AHB and peripherals
  - GPIO
  - AHB2 Master user extension interface
  - AHB2 Slave user extension interface
- AHB2APB Bridge
- APB and peripherals
  - UART0
  - UART1
  - Timer0
  - Timer1
  - Watch Dog
  - RTC
  - APB2 Extension Interface

1.2.2 FPGA Core System

FPGA core system includes:

- External crystal oscillator clock input or internal crystal oscillator clock can be as the system clock source of MCU core system. The max. frequency of the system clock is up to 200MHz (Subject to the project design and the chip in use).
- The reset signal input can be as the system reset signal of MCU core system.
- Six user interrupt handling signals for user extension peripherals
- AHB Extension interface
  - SRAM and FLASH can be as the data and instruction memory respectively of MCU core system
  - One AHB2 Master user extension interface
  - One AHB2 Slave user extension interface
- APB2 Extension Interface
  - SPI Master
  - I^2C Master
  - Twelve APB2 Master user extension interfaces
- Memory
  - The SRAM Size can be configured as 2KB, 4KB, 8KB and 16KB.
  - The FLASH Size as 32KB.

1.3 System Port

The definition of system ports is as shown in Table 1-1.

Table 1-1 Definition of System Ports

Name	I/O	Data Width	Description	Module
sys_clk	in	1	System clock signal	-
reset_n	in	1	System reset signal	-
trace_clk	out	1	TPIU clock signal	TPIU
trace_data	out	[3:0]	TPIU data output signal	
user_int_0	in	1	User Interrupt handling signal 0	NVIC
user_int_1	in	1	User Interrupt handling signal 1	

Name	I/O	Data Width	Description	Module
user_int_2	in	1	User Interrupt handling signal 2	
user_int_3	in	1	User Interrupt handling signal 3	
user_int_4	in	1	User Interrupt handling signal 4	
user_int_5	in	1	User Interrupt handling signal 5	
gpio	inout	[15:0]	GPIO signal	GPIO I/O
gpioin	in	[15:0]	GPIO input signal	GPIO non-I/O
gpioout	out	[15:0]	GPIO output signal	
gpioouten	out	[15:0]	GPIO output enable signal	
uart0_rxd	in	1	UART0 receive signal	UART0
uart0_txd	out	1	UART0 transmit signal	
uart1_rxd	in	1	UART1 receive signal	UART1
uart1_txd	out	1	UART1 transmit signal	
scl	inout	1	I^2C serial clock signal	I^2C Master I/O
sda	inout	1	I^2C serial data signal	
sclin	in	1	I^2C serial clock input signal	I^2C Master non-I/O
sclout	out	1	I^2C serial clock output signal	
sclouten	out	1	I^2C serial clock output enable signal	
sdain	in	1	I^2C serial data input signal	
sdaout	out	1	I^2C serial data output signal	
sdaouten	out	1	I^2C serial data output enable signal	
mosi	out	1	SPI master output / slave input	SPI Master
miso	in	1	SPI master input / slave output	
sclk	out	1	SPI clock signal	
nss	out	1	SPI slave selection signal	
rtc_src_clk	in	1	RTC signal	RTC
master_hclk	out	1	Master clock signal	AHB2 Master
master_hrst	out	1	Master reset signal	
master_hsel	out	1	Master selection signal	
master_haddr	out	[31:0]	Master address signal	
master_htrans	out	[1:0]	Master transmit type signal	

Name	I/O	Data Width	Description	Module
master_hwrite	out	1	I/O of Master read and write signal	
master_hsize	out	[2:0]	Master transmit Size signal	
master_hburst	out	[2:0]	Master burst type signal	
master_hprot	out	[3:0]	Master protect and control signal	
master_memattr	out	[1:0]	Master memattr signal	
master_exreq	out	1	Matter exreq signal	
master_hmaster	out	[3:0]	Master label signal	
master_hwdata	out	[31:0]	Master write data signal	
master_hmastlock	out	1	Master lock signal	
master_hreadymux	out	1	Master hreadymux signal	
master_hauser	out	1	Master hauser signal	
master_hwuser	out	[3:0]	Master hwuser signal	
master_hrdata	in	[31:0]	Master read data signal	
master_hreadyout	in	1	Master hreadyout signal	
master_hresp	in	1	Master transmit status signal	
master_exresp	in	1	Master exresp signal	
master_hruser	in	[2:0]	Master hruser signal	
slave_hsel	in	1	Slave selection signal	AHB2 Slave
slave_haddr	in	[31:0]	Slave address signal	
slave_htrans	in	[1:0]	Slave transmit type signal	
slave_hwrite	in	1	I/O of Slave read and write signal	
slave_hsize	in	[2:0]	Slave transmit Size signal	
slave_hburst	in	[2:0]	Slave burst type signal	
slave_hprot	in	[3:0]	Slave protect and control signal	
slave_hmaster	in	[3:0]	Slave label signal	
slave_hwdata	in	[31:0]	Slave write data signal	
slave_hmastlock	in	1	Slave lock signal	
slave_hrdata	out	[31:0]	Slave read data signal	
slave_hready	out	1	Slave ready signal	
slave_hresp	out	1	Slave transmit status signal	
slave_hexresp	out	1	Slave hexresp signal	

Name	I/O	Data Width	Description	Module
slave_hruser	out	[2:0]	Slave hruser signal	
slave_hmemattr	in	[1:0]	Slave hmemattr signal	
slave_hexreq	in	1	Slave hexreq signal	
slave_hauser	in	1	Slave hauser signal	
slave_hwuser	in	[3:0]	Slave hwuser signal	
master_pclk	out	1	APB2 Master clock signal	APB2 Master [1-12]
master_prst	out	1	APB2 Master reset signal	
master_penable	out	1	APB2 Master enable signal	
master_paddr	out	[7:0]	APB2 Master address signal	
master_pwrite	out	1	I/O of APB2 Master read and write signal	
master_pwdata	out	[31:0]	APB2 Master write data signal	
master_pstrb	out	[3:0]	APB2 Master write strobe signal	
master_pprot	out	[2:0]	APB2 Master protect type signal	
master_psel1	out	1	APB2 Master [1] selection signal	APB2 Master [1]
master_pready1	in	1	APB2 Master [1] ready Signal	
master_prdata1	in	[31:0]	APB2 Master [1] read data signal	
master_pslverr1	in	1	APB2 Master[ ] transmit failure signal	
master_psel2	out	1	APB2 Master [2] selection signal	APB2 Master [2]
master_pready2	in	1	APB2 Master [2] ready Signal	
master_prdata2	in	[31:0]	APB2 Master [2] read data signal	
master_pslverr2	in	1	APB2 Master[ ] transmit failure signal	
master_psel3	out	1	APB2 Master [3] selection signal	APB2 Master [3]
master_pready3	in	1	APB2 Master [3] ready Signal	
master_prdata3	in	[31:0]	APB2 Master [3] read data signal	
master_pslverr3	in	1	APB2 Master[ ] transmit failure signal	
master_psel4	out	1	APB2 Master [4] selection signal	APB2 Master [4]
master_pready4	in	1	APB2 Master [4] ready signal	

Name	I/O	Data Width	Description	Module
master_prdata4	in	[31:0]	APB2 Master [4] read data signal	
master_pslverr4	in	1	APB2 Master[ ] transmit failure signal	
master_psel5	out	1	APB2 Master [5] selection signal	APB2 Master [5]
master_pready5	in	1	APB2 Master [5] ready signal	
master_prdata5	in	[31:0]	APB2 Master [5] read data signal	
master_pslverr5	in	1	APB2 Master[ ] transmit failure signal	
master_psel6	out	1	APB2 Master [6] selection signal	APB2 Master [6]
master_pready6	in	1	APB2 Master [6] ready signal	
master_prdata6	in	[31:0]	APB2 Master [6] read data signal	
master_pslverr6	in	1	APB2 Master[ ] transmit failure signal	
master_psel7	out	1	APB2 Master [7] selection signal	APB2 Master [7]
master_pready7	in	1	APB2 Master [7] ready signal	
master_prdata7	in	[31:0]	APB2 Master [7] read data signal	
master_pslverr7	in	1	APB2 Master[ ] transmit failure signal	
master_psel8	out	1	APB2 Master [8] selection signal	APB2 Master [8]
master_pready8	in	1	APB2 Master [8] ready signal	
master_prdata8	in	[31:0]	APB2 Master [8] read data signal	
master_pslverr8	in	1	APB2 Master[ ] transmit failure signal	
master_psel9	out	1	APB2 Master [9] selection signal	APB2 Master [9]
master_pready9	in	1	APB2 Master [9] ready signal	
master_prdata9	in	[31:0]	APB2 Master [9] read data signal	
master_pslverr9	in	1	APB2 Master[ ] transmit failure signal	
master_psel10	out	1	APB2 Master [10] selection	APB2 Master [10]

Name	I/O	Data Width	Description	Module
			signal	
master_pready10	in	1	APB2 Master [10] ready signal	
master_prdata10	in	[31:0]	APB2 Master [10] read data signal	
master_pslverr10	in	1	APB2 Master[ ] transmit failure signal	
master_psel11	out	1	APB2 Master [11] selection signal	APB2 Master [11]
master_pready11	in	1	APB2 Master [11] ready signal	
master_prdata11	in	[31:0]	APB2 Master [11] read data signal	
master_pslverr11	in	1	APB2 Master[11] transmit failure signal	
master_psel12	out	1	APB2 Master [12] selection signal	APB2 Master [12]
master_pready12	in	1	APB2 Master [12] ready signal	
master_prdata12	in	[31:0]	APB2 Master [12] read data signal	
master_pslverr12	in	1	APB2 Master[12] transmit failure signal	

1.4 System Resource Utilization and Performance Statistics

The system resource utilization and performance statistics of Gowin_EMPU(GW1NS-4C) are as shown in Table 1-2 .

Table 1-2 System Resource Utilization and Performance Statistics

Resources / Configuration	LUTs	Registers	BSRAMs	Frequency (MHz)
MCU core system	33	19	4	200
MCU core system + I2C Master + SPI Master	456	282	4	100
MCU core system + APB SPI-Flash Memory	1073	618	4	50
MCU core system + AHB PSRAM Memory	1709	1464	4	100
MCU core ssystem + AHB HyperRAM Memory	1163	1008	4	100

2 Hardware Design Flow

2.1 Hardware Target

- DK-START-GW1NSR4C-QN48G V1.1
  GW1NSR-LV4CQN48GC7/I6
- DK-START-GW1NSR4C-QN48P V1.1
  GW1NSR-LV4CQN48PC7/I6
- DK-START-GW1NSR4C-MG64P V1.1
  GW1NSR-LV4CMG64PC7/I6

2.2 Software Version

Tested software version: Gowin_V1.9.9.01 (64-bit)

2.3 IP Core Generator

The IP Core Generator tool from Gowin Software is used to configure and generate Gowin_EMPU(GW1NS-4C) IP Core.

2.4 Programmer

Gowin Programmer tool is used to download the bitstream file of Gowin_EMPU(GW1NS-4C).

For the Gowin Programmer usage, please refer to *SUG502, Gowin Programmer User Guide*.

2.5 Design Flow

Gowin_EMPU(GW1NS-4C) hardware design flow is as follows:

- Configure and generate Gowin_EMPU(GW1NS-4C) IP Core using IP

Core Generator and import the current project.

- Instantiate Gowin_EMPU(GW1NS-4C), import user design, and connect user design with Gowin_EMPU Top Module.
- Add physical and timing Constraints.
- Use GowinSynthesis to synthesize and generate post-synthesis netlist file.
- Use Place & Route to generate bitstream file.
- Use Programmer to download bitstream file to chip to chip.

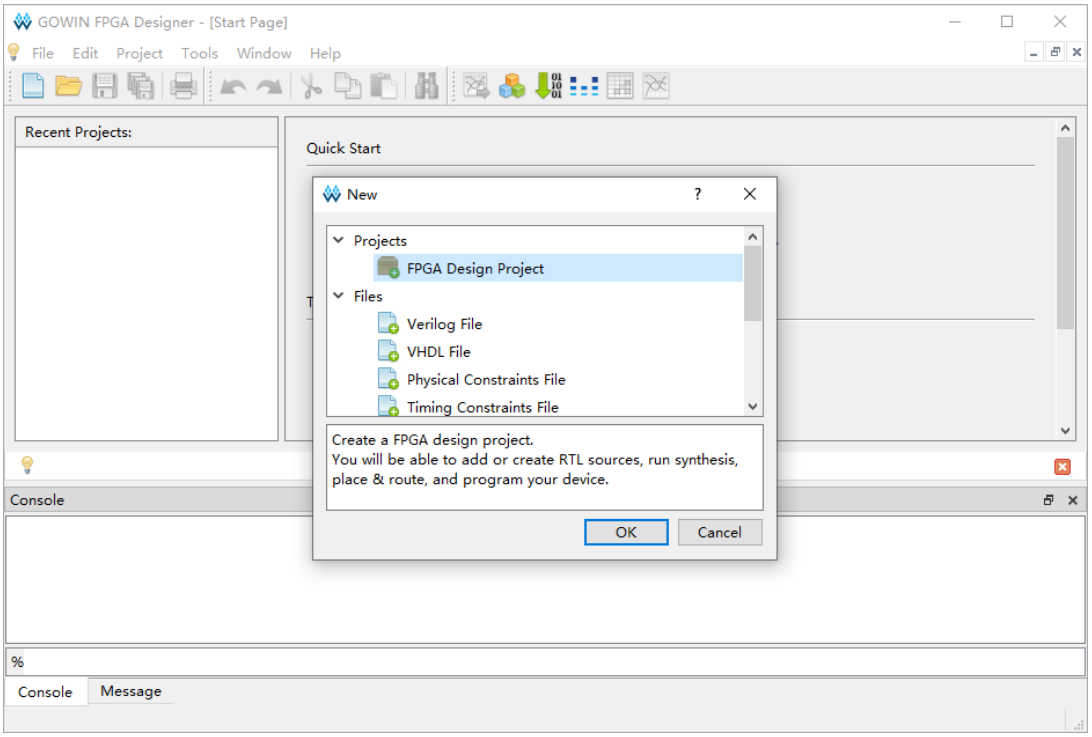
3 Project Template

3.1 Project Creation

3.1.1 Create a New Project

Double click to open the Gowin Software. Click "File > New… > FPGA Design Project" on the menu bar, as shown in Figure 3-1.

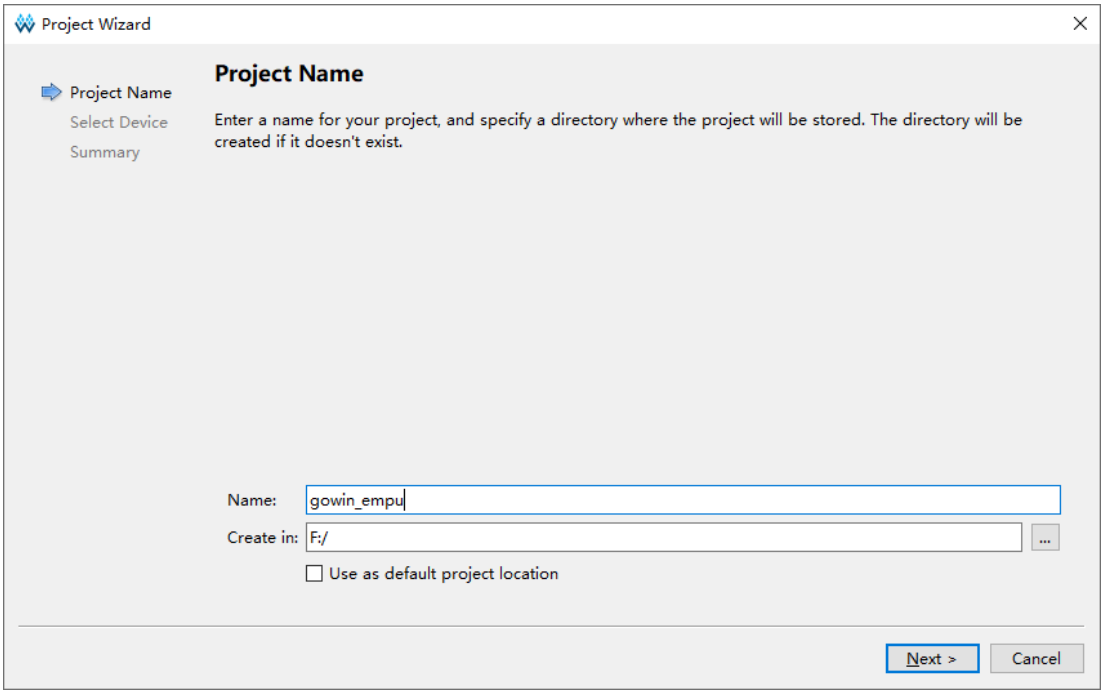
Figure 3-1 Create a FPGA Design Project



3.1.2 Set Project Name and Path

Enter the project name and select the project path, as shown in Figure 3-2.

Figure 3-2 Set Project Name and Path



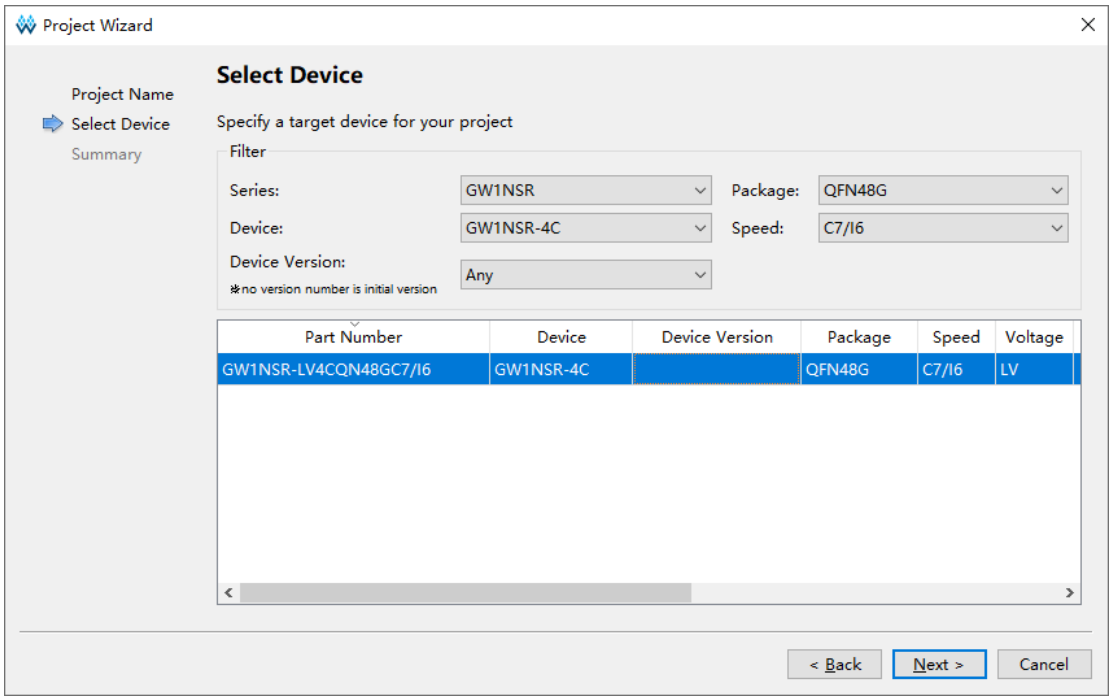
3.1.3 Select Device

Configure "Series", "Device", "Device Version", "Package", "Speed", and "Part Number", as shown in Figure 3-3.

Take DK_START_GW1NSR4C_QN48G_V1.1 reference design as an instance.

- Series: GW1NSR
- Device: GW1NSR-4C
- Device Version: Any
- Package: QFN48G
- Speed: C7/I6
- Part Number: GW1NSR-LV4CQN48GC7/I6

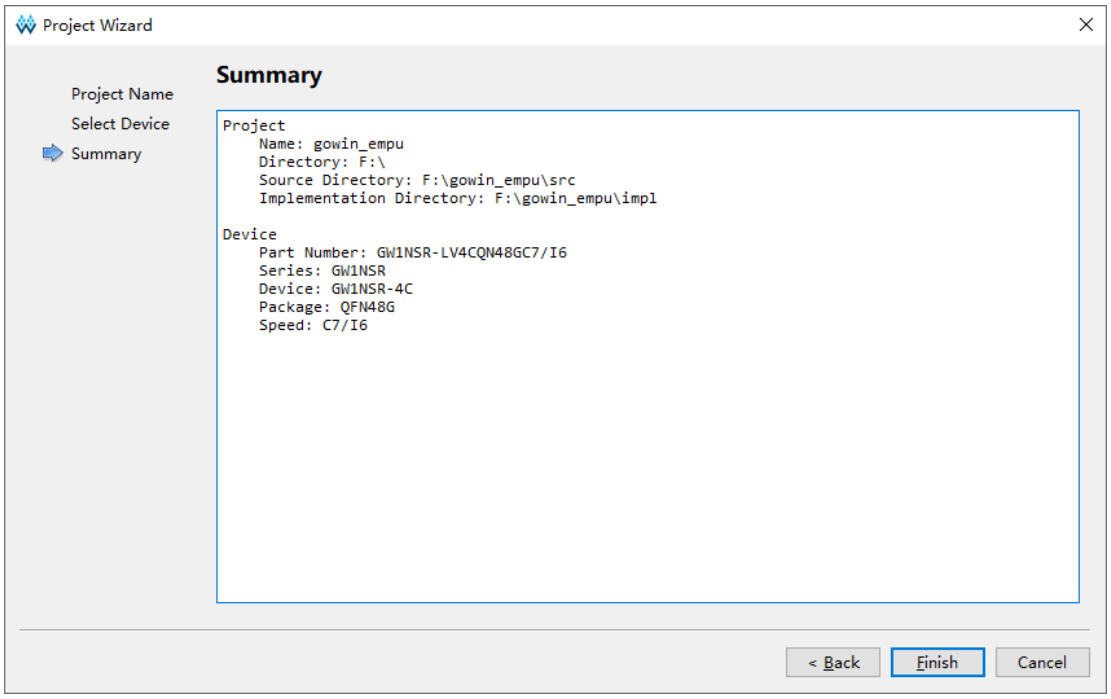
Figure 3-3 Select Device



3.1.4 Project Creation Completed

Project creation is completed, as shown in Figure 3-4.

Figure 3-4 Project Creation Completed



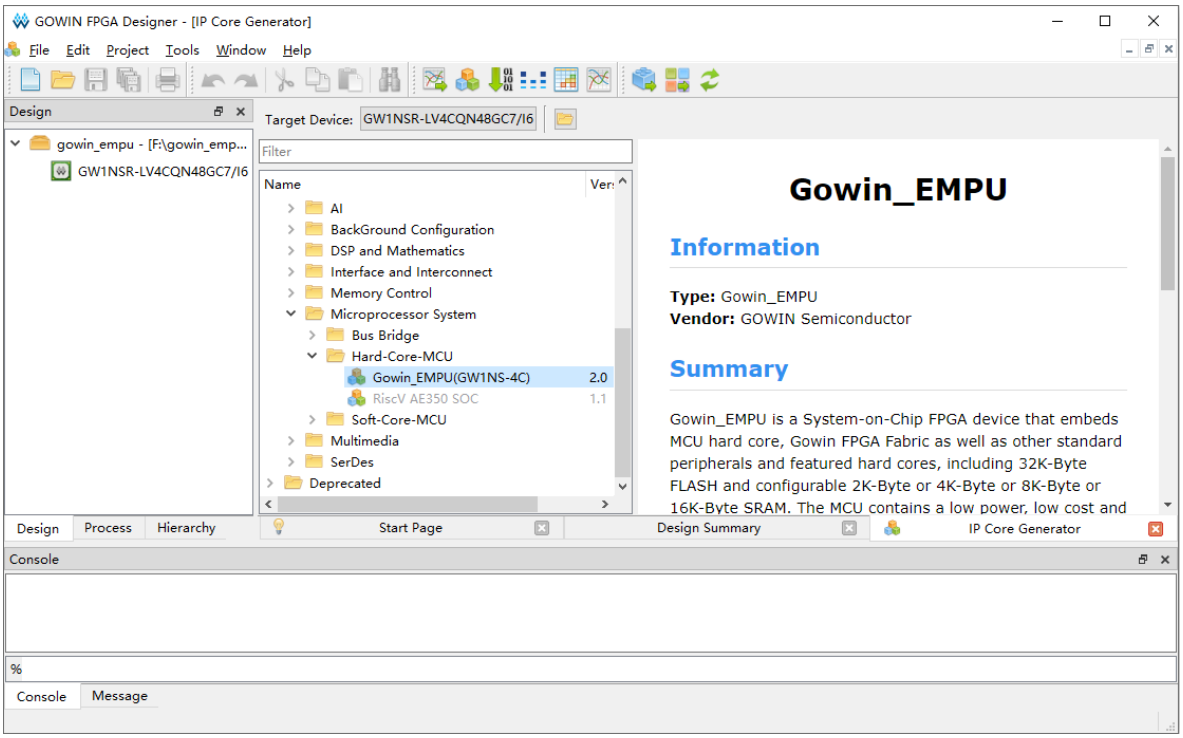
3.2 Hardware Design

Use IP Core Generator to generate Gowin_EMPU(GW1NS-4C) hardware design.

Select "Tools > IP Core Generator" in the menu bar or "  " to open the IP Core Generator.

Select " Soft IP Core > Microprocessor System > Hard-Core-MCU > Gowin_EMPU(GW1NS-4C) 2.0", as shown in Figure 3-5.

Figure 3-5 Select Gowin_EMPU(GW1NS-4C) IP Core



The system architecture of Gowin_EMPU(GW1NS-4C) is as shown in Figure 3-6.

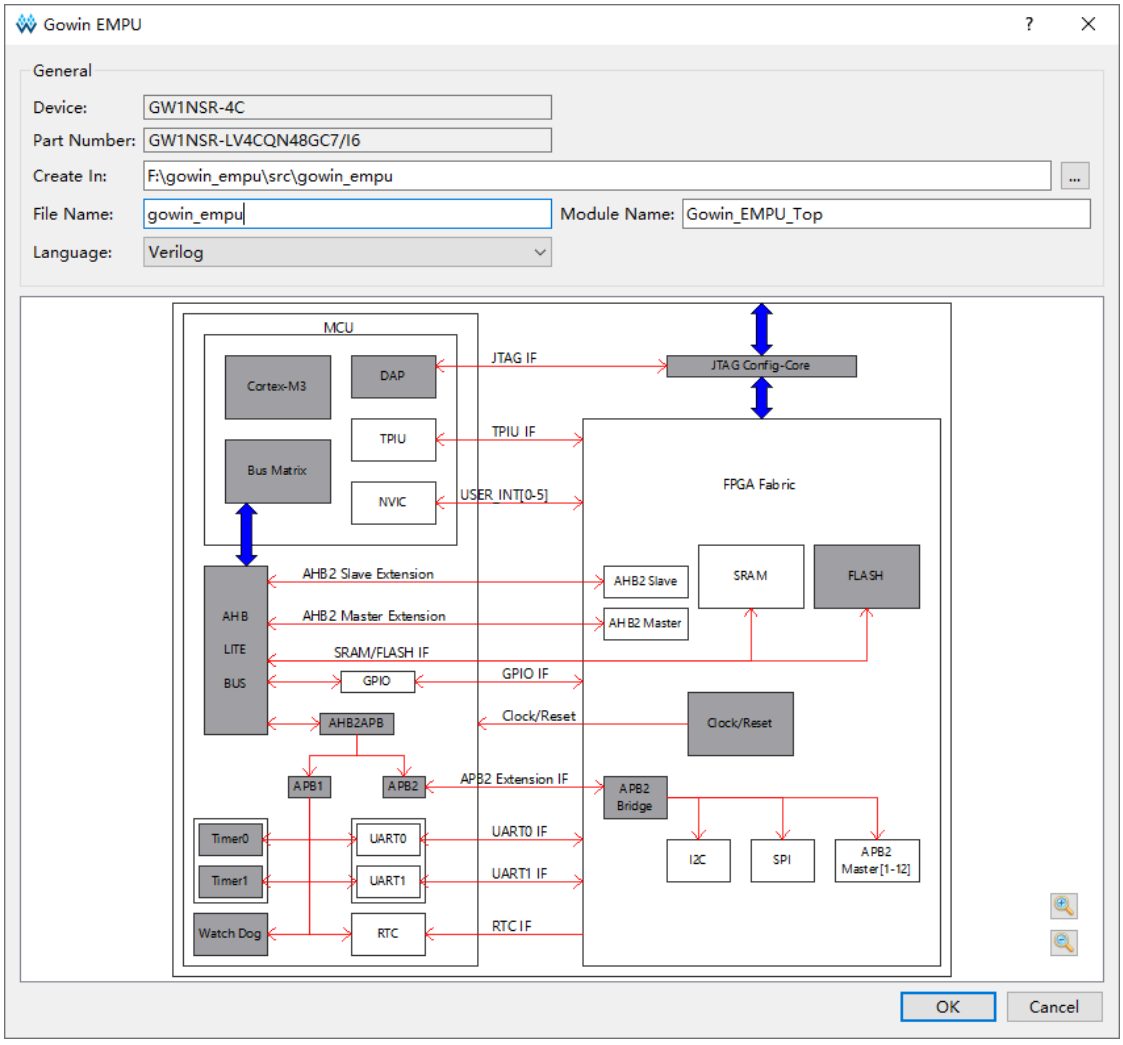
The grayed modules are the default, and you can not configure; If the modules are not grayed, you can open the modules to configure.

The modules that you can choose to configure include:

- TPIU
- NVIC: six user interrupt handling signals USER_INT0~5
- AHB2 Slave: FPGA core system can extend AHB2 Slave peripherals.
- AHB2 Master: FPGA core system can extend AHB2 Master peripherals.

- GPIO
- UART0 and UART1
- RTC
- The SRAM can be configured as 2KB, 4KB, 8KB or 16KB, 16KB by default.
- I²C: FPGA core system integrates I²C Master.
- SPI: FPGA core system integrates SPI Master.
- APB2 Master[1 -12]: FPGA core system can extend twelve APB2 Master user devices.

Figure 3-6 System Architecture



The system configuration options of Gowin_EMPU(GW1NS-4C) are as shown in Table 3-1.

Table 3-1 System Configuration Options

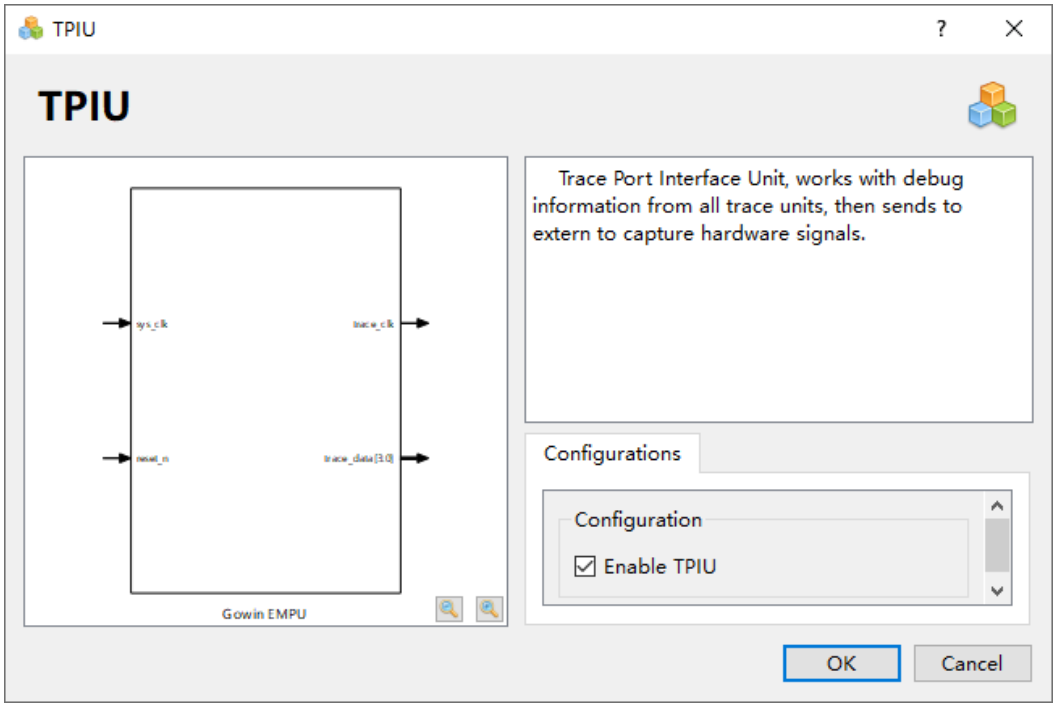
Options	Description
Enable TPIU	Enable TPIU, disabled by default
Enable USER_INT_0	Enable user interrupt handling signal [0], disabled by default.
Enable USER_INT_1	Enable user interrupt handling signal [1], disabled by default.
Enable USER_INT_2	Enable user interrupt handling signal [2], disabled by default.
Enable USER_INT_3	Enable user interrupt handling signal [3], disabled by default.
Enable USER_INT_4	Enable user interrupt handling signal [4], disabled by default.
Enable USER_INT_5	Enable user interrupt handling signal [5], disabled by default.
Enable GPIO	Enable GPIO, disabled by default.
Enable GPIO I/O	Enable GPIO inout port, enabled by default.
Enable UART0	Enable UART0, disabled by default.
Enable UART1	Enable UART1, disabled by default.
Enable RTC	Enable RTC, disabled by default.
Enable AHB2 Master	Enable AHB2 Master user extension interface, disabled by default.
Enable AHB2 Slave	Enable AHB2 Slave user extension interface, disabled by default.
Enable I^2C	Enable I^2C Master, disabled by default.
Enable I^2C I/O	Enable I^2C inout port, enabled by default.
Enable SPI	Enable SPI Master, disabled by default.
Enable APB2 Master 1	Enable APB2 Master [1] user extension interface, disabled by default.
Enable APB2 Master 2	Enable APB2 Master [2] user extension interface, disabled by default.
Enable APB2 Master 3	Enable APB2 Master [3] user extension interface, disabled by default.
Enable APB2 Master 4	Enable APB2 Master [4] user extension interface, disabled by default.
Enable APB2 Master 5	Enable APB2 Master [5] user extension interface, disabled by default.
Enable APB2 Master 6	Enable APB2 Master [6] user extension interface, disabled by default.
Enable APB2 Master 7	Enable APB2 Master [7] user extension interface, disabled by default.
Enable APB2 Master 8	Enable APB2 Master [8] user extension interface, disabled by default.
Enable APB2 Master 9	Enable APB2 Master [9] user extension interface, disabled by default.
Enable APB2 Master 10	Enable APB2 Master [10] user extension interface, disabled by default.
Enable APB2 Master 11	Enable APB2 Master [11] user extension interface, disabled by default.
Enable APB2 Master 12	Enable APB2 Master [12] user extension interface, disabled by default.
SRAM Size	The SRAM Size can be configured as 2KB, 4KB, 8KB and 16KB, 16KB by default.

3.2.1 TPIU Configuration

Double click to configure TPIU, as shown in Figure 3-7.

If "Enable TPIU" is selected, Gowin_EMPU(GW1NS-4C) supports TPIU, disabled by default.

Figure 3-7 TPIU Configuration



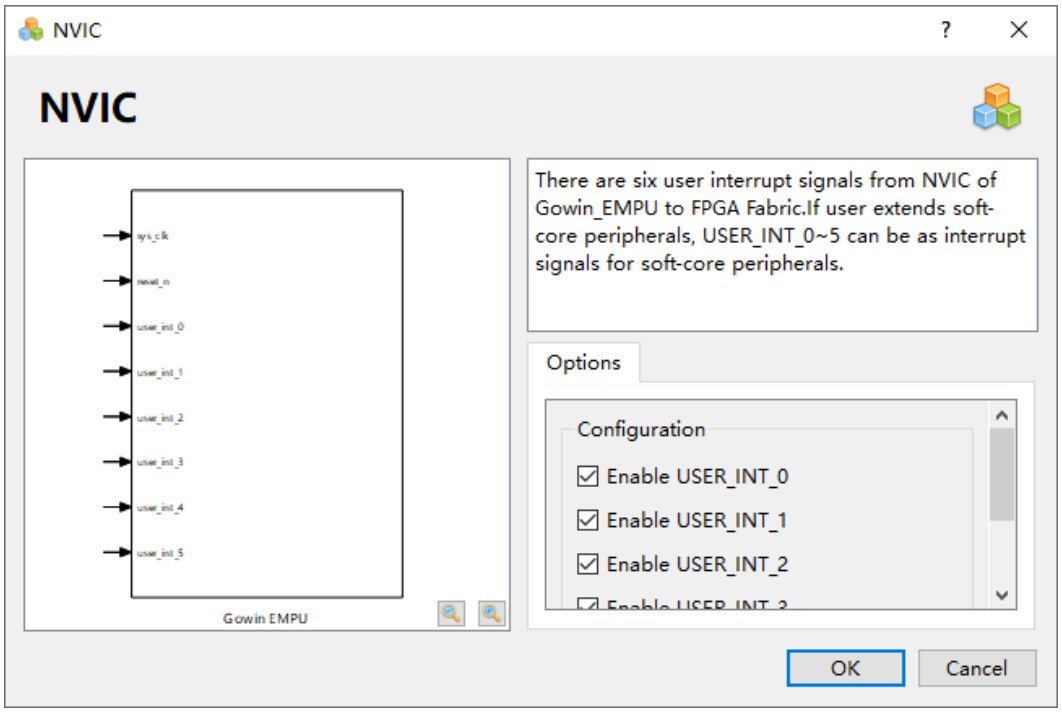
3.2.2 NVIC Configuration

Double click to open NVIC to configure USER_INT_0~5 for user extension peripherals of FPGA core system, as shown in Figure 3-8.

- If Enable USER_INT_0 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [0], disabled by default.
- If Enable USER_INT_1 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [1], disabled by default.
- If Enable USER_INT_2 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [2], disabled by default.
- If Enable USER_INT_3 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [3], disabled by default.
- If Enable USER_INT_4 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [4], disabled by default.

- If Enable USER_INT_5 is selected, Gowin_EMPU(GW1NS-4C) supports user interrupt handling signal [5], disabled by default.

Figure 3-8 NVIC Configuration

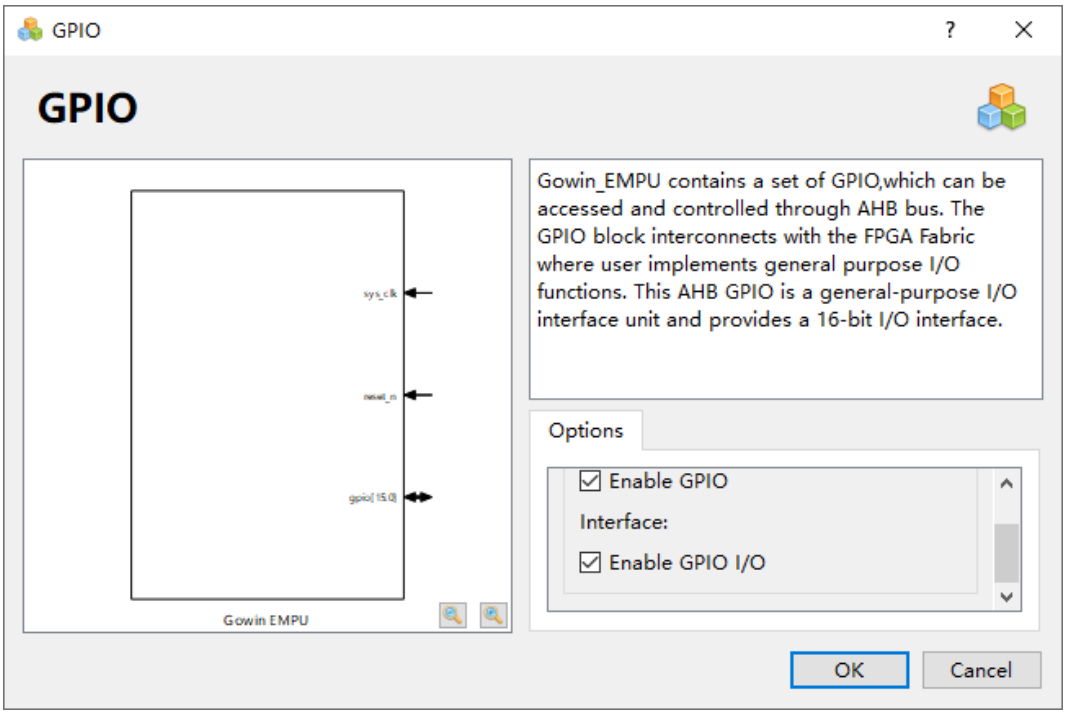


3.2.3 GPIO Configuration

Double click to configure GPIO, as shown in Figure 3-9.

- If "Enable GPIO" is selected, Gowin_EMPU(GW1NS-4C) supports GPIO, disabled by default.
- If "Enable GPIO" is selected, you can configure GPIO port type.
- If "Enable GPIO I/O" is selected, GPIO supports inout port, supported by default.

Figure 3-9 GPIO Configuration

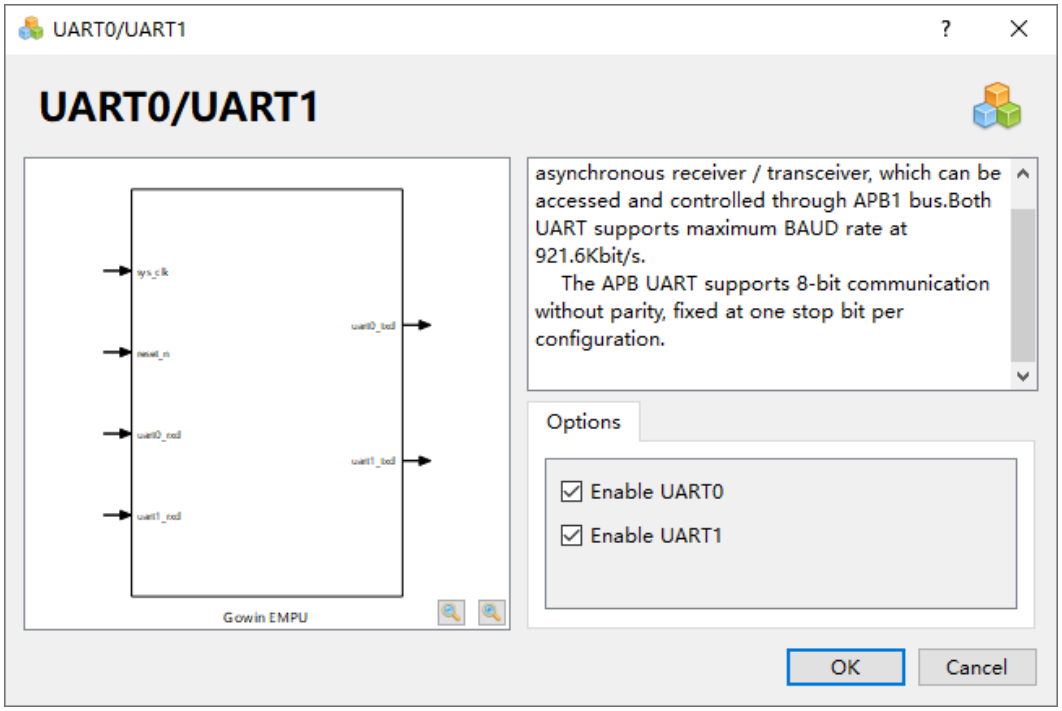


3.2.4 UART Configuration

Double click to configure UART0 or UART1, as shown in Figure 3-10.

- If "Enable UART0" is selected, Gowin_EMPU(GW1NS-4C) supports UART0, disabled by default.
- If "Enable UART1" is selected, Gowin_EMPU(GW1NS-4C) supports UART1, disabled by default.

Figure 3-10 UART Configuration

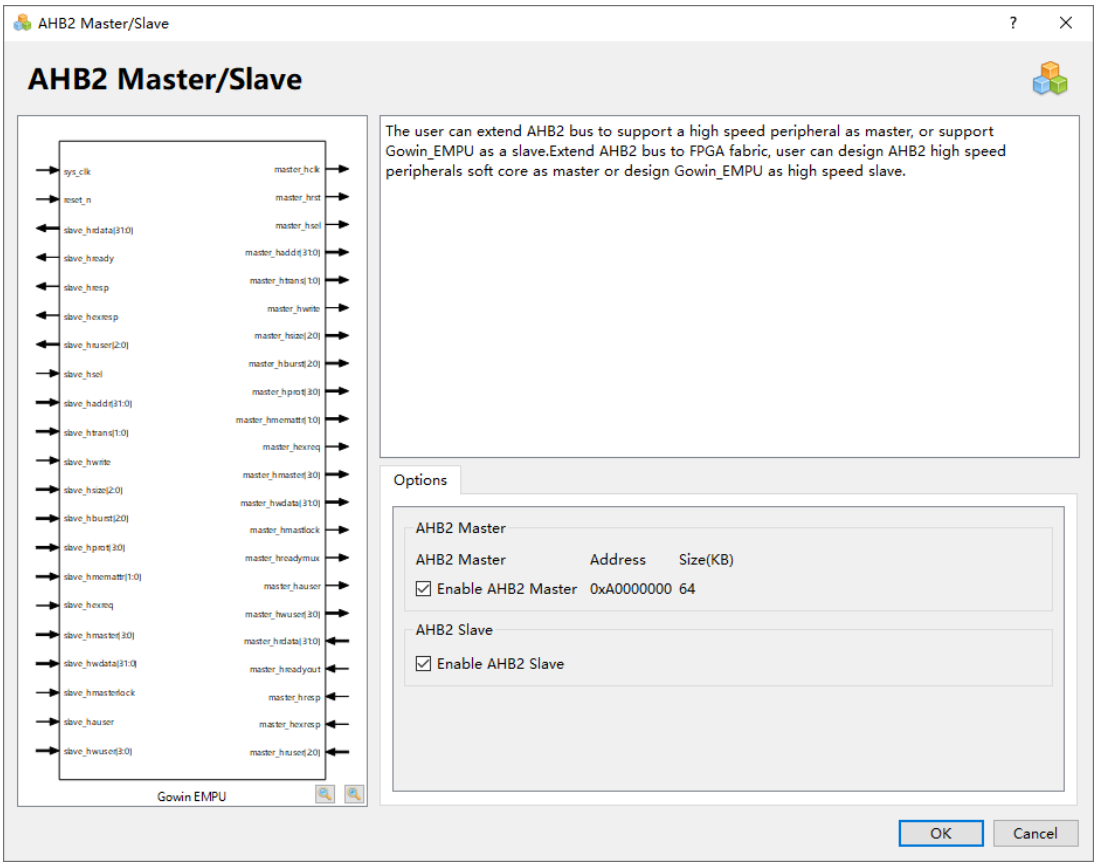


3.2.5 AHB2 Master / Slave Configuration

Double click to configure AHB2 Master or AHB2 Slave, as shown in Figure 3-11.

- If "Enable AHB2 Master" is selected, Gowin_EMPU(GW1NS-4C) supports AHB2 Master, disabled by default.
- If "Enable AHB2 Slave" is selected, Gowin_EMPU(GW1NS-4C) supports AHB2 Slave, disabled by default.
- The base address mapping of AHB2 Master user extension peripherals is 0xA0000000.

Figure 3-11 AHB2 Master/Slave Configuration

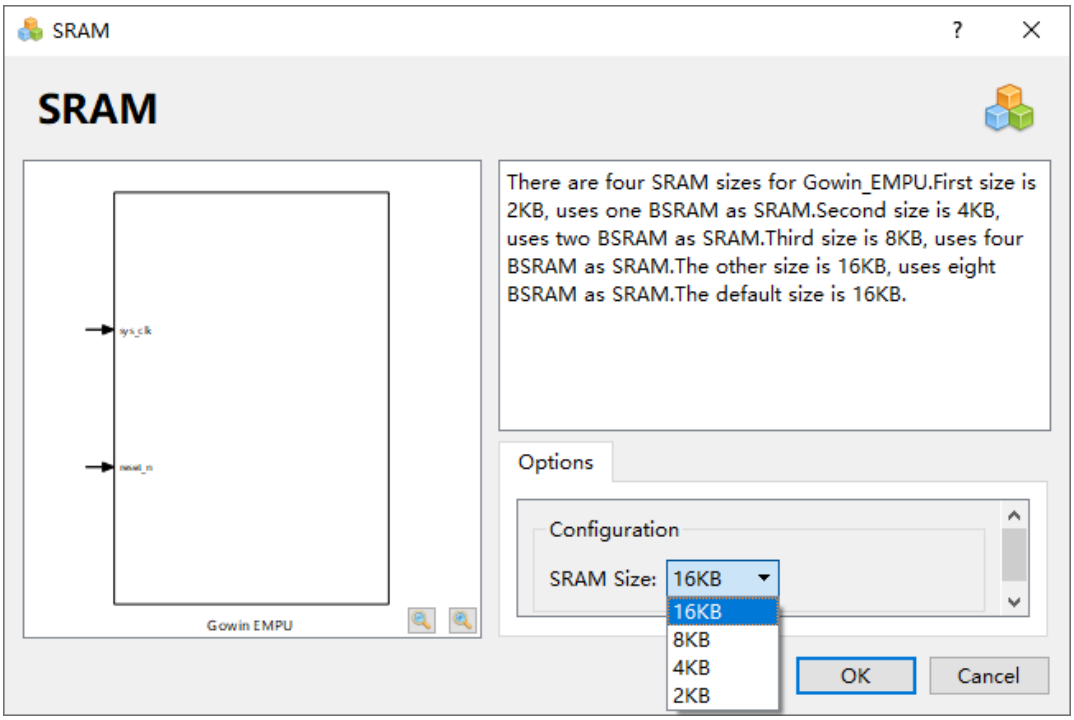


3.2.6 SRAM Configuration

Double click to configure SRAM Size, as shown in Figure 3-12.

The SRAM Size can be configured as 2KB, 4KB, 8KB or16KB, 16KB by default.

Figure 3-12 SRAM Configuration

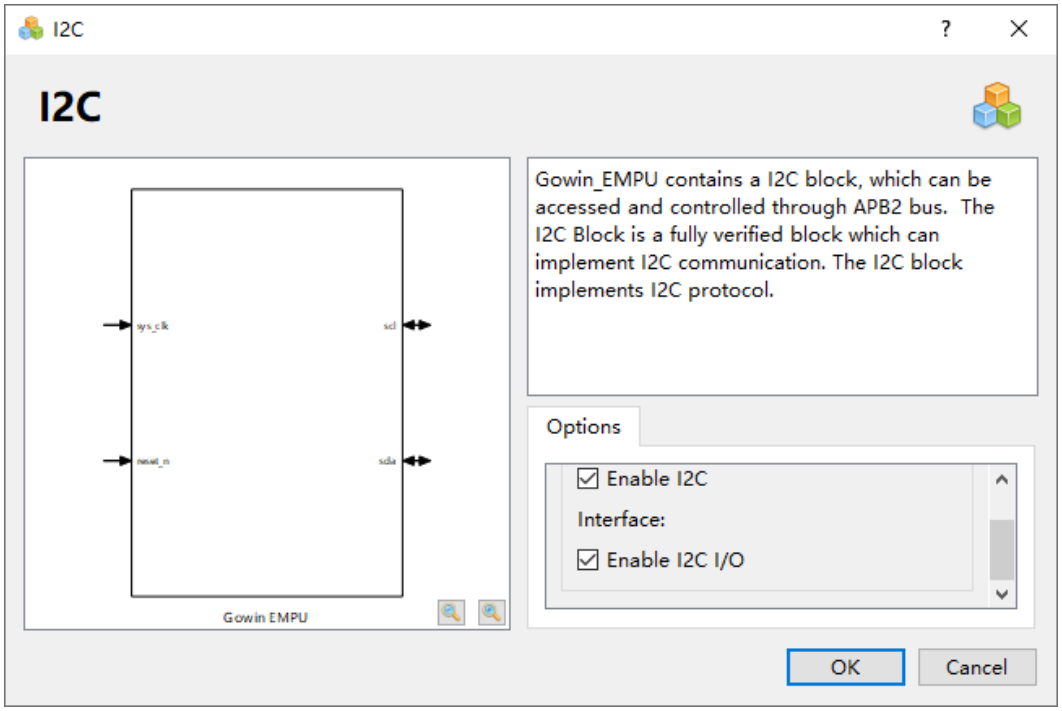


3.2.7 I²C Configuration

Double click to configure I²C Master, as shown in Figure 3-13.

- If "Enable I²C" is selected, Gowin_EMPU(GW1NS-4C) supports I²C Master, disabled by default.
- If "Enable I2C" is selected, you can configure I²C Master port type.
- If "Enable I2C I/O" is selected, I²C Master supports inout port, supported by default.

Figure 3-13 I²C Configuration

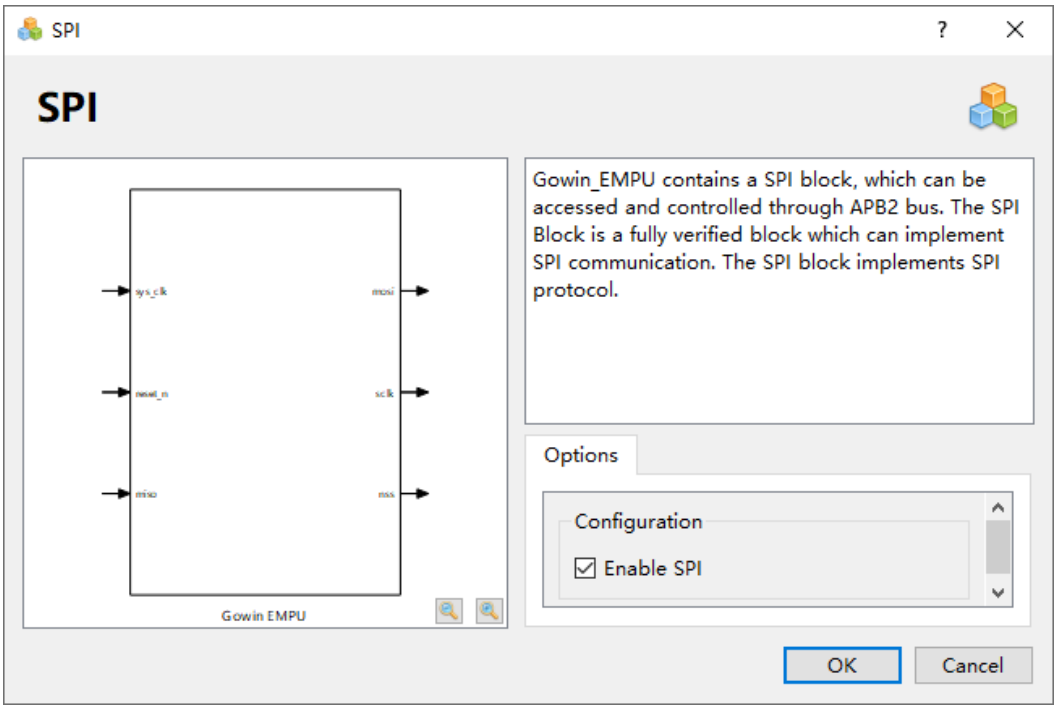


3.2.8 SPI Configuration

Double click to configure SPI Master, as shown in Figure 3-14.

If "Enable SPI" is selected, Gowin_EMPU(GW1NS-4C) supports SPI Master, disabled by default.

Figure 3-14 SPI Configuration



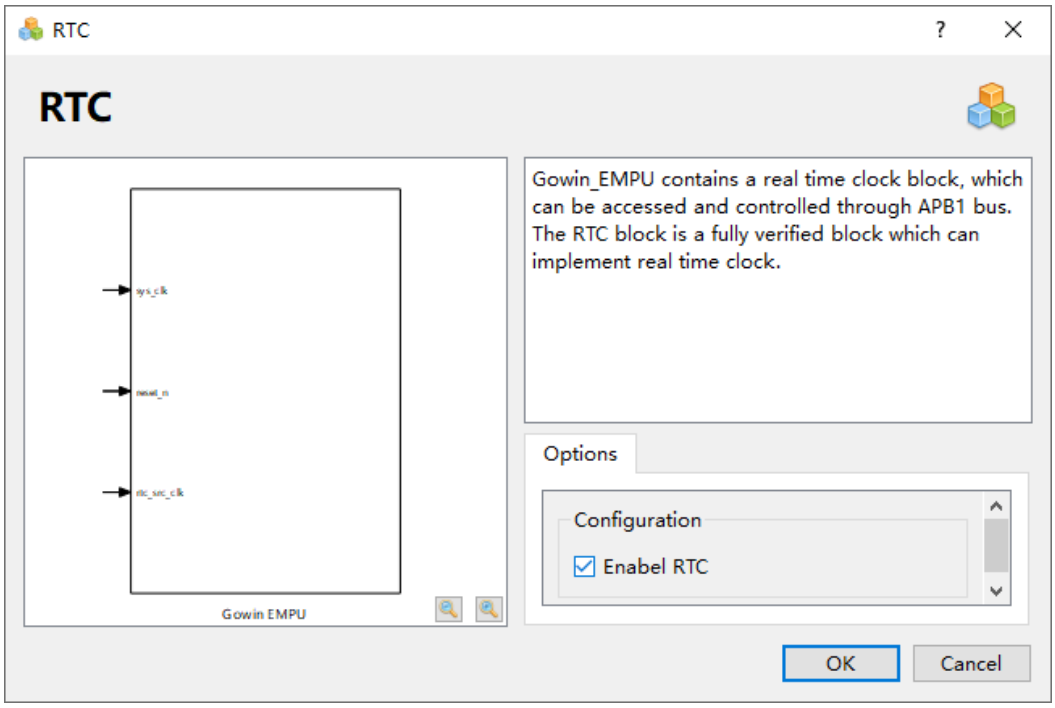
3.2.9 RTC Configuration

Double click to configure RTC, as shown in Figure 3-15.

If "Enable RTC" is selected, Gowin_EMPU(GW1NS-4C) supports RTC, disabled by default.

The port rtc_src_clk is input a 3.072MHz clock, internally divided by the RTC to 1Hz.

Figure 3-15 RTC Configuration



3.2.10 APB2 Master Configuration

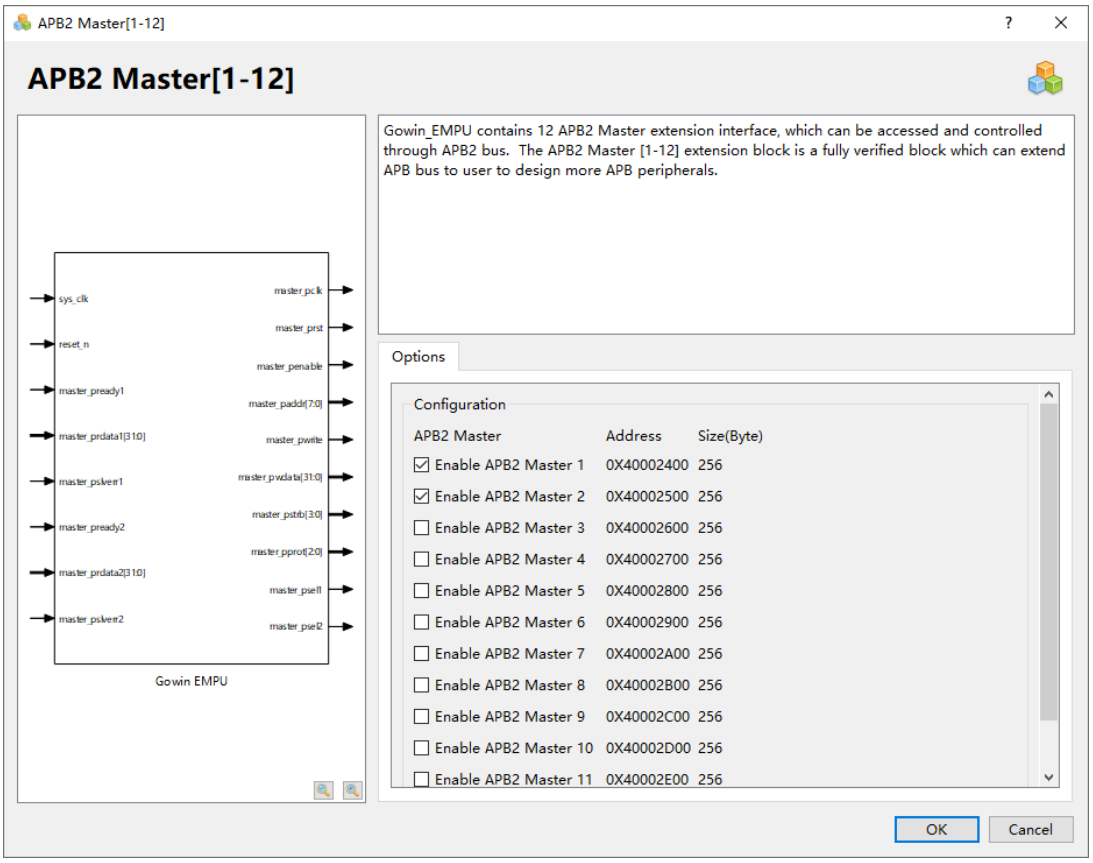
Double click to configure APB2 Master[1 -12], as shown in Figure 3-16.

- If "Enable APB2 Master 1" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [1], disabled by default.
- If "Enable APB2 Master 2" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [2], disabled by default.
- If "Enable APB2 Master 3" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [3], disabled by default.
- If "Enable APB2 Master 4" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [4], disabled by default.
- If "Enable APB2 Master 5" is selected, Gowin_EMPU(GW1NS-4C)

supports APB2 Master [5], disabled by default.

- If "Enable APB2 Master 6" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [6], disabled by default.
- If "Enable APB2 Master 7" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [7], disabled by default.
- If "Enable APB2 Master 8" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [8], disabled by default.
- If "Enable APB2 Master 9" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [9], disabled by default.
- If "Enable APB2 Master 10" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [10], disabled by default.
- If "Enable APB2 Master 11" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [11], disabled by default.
- If "Enable APB2 Master 12" is selected, Gowin_EMPU(GW1NS-4C) supports APB2 Master [12], disabled by default.

Figure 3-16 APB2 Master[1-12] Configuration



The base address mapping of APB2 Master [1-12] user extension peripherals is as shown in Table 3-2.

Table 3-2 Base Address Mapping of APB2 Master [1-12]

APB2 Master	Address	Size(Byte)
1	0x40002400	256
2	0x40002500	256
3	0x40002600	256
4	0x40002700	256
5	0x40002800	256
6	0x40002900	256
7	0x40002A00	256
8	0x40002B00	256
9	0x40002C00	256
10	0x40002E00	256
11	0x40002E00	256
12	0x40002F00	256

3.2.11 PSRAM Memory Interface

If FPGA product GW1NSR-4C MG64P is selected, Gowin_EMPU (GW1NS-4C) supports the FPGA core system extended external device PSRAM Memory Interface.

The software development kit provides the external device PSRAM Memory Interface as a reference design.

Hardware Design Flow

- IP Core Generator is used to configure and generate Gowin_EMPU (GW1NS-4C) IP Core, and the AHB2 Master user extension interface is enabled.
- IP Core Generator is used to configure and generate PSRAM Memory Interface IP Core.
  - Memory Clock 100MHz
  - For other options, you can select default configuration.
- Design the AHB PSRAM Memory Interface IP implementing the AHB bus interface, refer to gw_ahb_psram.v.
- Instantiate Gowin_EMPU(GW1NS-4C) Top Module and AHB PSRAM Memory Interface Top Module.
- Connect Gowin_EMPU(GW1NS-4C) to AHB interface of AHB PSRAM

Memory Interface.

Reference Design

You can click here to get the following reference design:

- Hardware Reference Design …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_MG64P_V1.1\gowin_empu\cm3_psram_demo
- Software Programming Reference Design
  - …\ref_design\MCU_RefDesign\MDK_RefDesign\cm3_demo\project\psram
  - …\ref_design\MCU_RefDesign\GMD_RefDesign\cm3_demo\project\psram

3.2.12 HyperRAM Memory Interface

If FPGA product GW1NSR-4C or GW1NSER-4C QN48P is selected, Gowin_EMPU (GW1NS-4C) supports the FPGA core system extended external device HyperRAM Memory Interface.

The software development kit provides the external device HyperRAM Memory Interface as a reference design.

Hardware Design Flow

- IP Core Generator is used to configure and generate Gowin_EMPU (GW1NS-4C), and AHB2 Master user extension interface is enabled.
- IP Core Generator is used to configure and generate HyperRAM Memory Interface embedded IP Core.
  - Memory Clock 100MHz
  - For other options, you can select default configuration.
- Design AHB HyperRAM Memory Interface IP implementing the AHB bus interface, refer to gw_ahb_hyperram.v.
- Instantiate Gowin_EMPU(GW1NS-4C) Top Module and AHB HyperRAM Memory Interface Top Module.
- Connect Gowin_EMPU(GW1NS-4C) to AHB interface of AHB HyperRAM Memory Interface.

Reference Design

You can click here to get the following reference design:

- Hardware Reference Design
  …

\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48P_V1.1\gowin_empu\cm3_hyperram_demo

- Software Programming Reference Design
  - …\ref_design\MCU_RefDesign\MDK_RefDesign\cm3_demo\project\hyper_ram
  - …\ref_design\MCU_RefDesign\GMD_RefDesign\cm3_demo\project\hyper_ram

3.2.13 SPI-Flash Memory

If FPGA product GW1NSR-4C or GW1NSER-4C QN48G is selected, Gowin_EMPU (GW1NS-4C) supports the FPGA core system extended external device SPI-Flash Memory.

The software development kit provides the external device SPI-Flash Memory as a reference design.

Hardware Design Flow

- You can configure and generate Gowin_EMPU (GW1NS-4C) IP Core using IP Core Generator, and APB2 Master [1] user extension interface is enabled.
- Design SPI-Flash Memory IP, refer to gw_spiflash.v.
- Design APB SPI-Flash Memory IP implementing the APB bus interface, refer to gw_spiflash.v.
- Instantiate Gowin_EMPU(GW1NS-4C) Top Module and APB SPI-Flash Memory Top Module.
- Connect Gowin_EMPU(GW1NS-4C) to APB interface of APB SPI-Flash Memory.

Reference Design

You can click here to get the following reference design:

- Hardware Reference Design … \ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48G_V1.1\gowin_empu\cm3_spiflash_demo
- Software Programming Reference Design
  - …\ref_design\MCU_RefDesign\MDK_RefDesign\cm3_demo\project\spi_flash
  - …\ref_design\MCU_RefDesign\GMD_RefDesign\cm3_demo\project\spi_flash

3.3 User Design

- After configuration, you can generate Gowin_EMPU(GW1NS-4C) IP Core.
- Instantiate Gowin_EMPU(GW1NS-4C) Top Module.
- Import user design and connect it with Gowin_EMPU(GW1NS-4C) Top Module to form a complete RTL design.

3.4 Constraint

After RTL design is completed, physical constraints can be generated according to the used development board and the IO.

Timing constraints file can be generated according to timing requirements.

For the details on physical constraints generation, please refer to *SUG940, Gowin Design Timing Constraints Guide*, *SUG935, Gowin Design Physical Constraints Guide*.

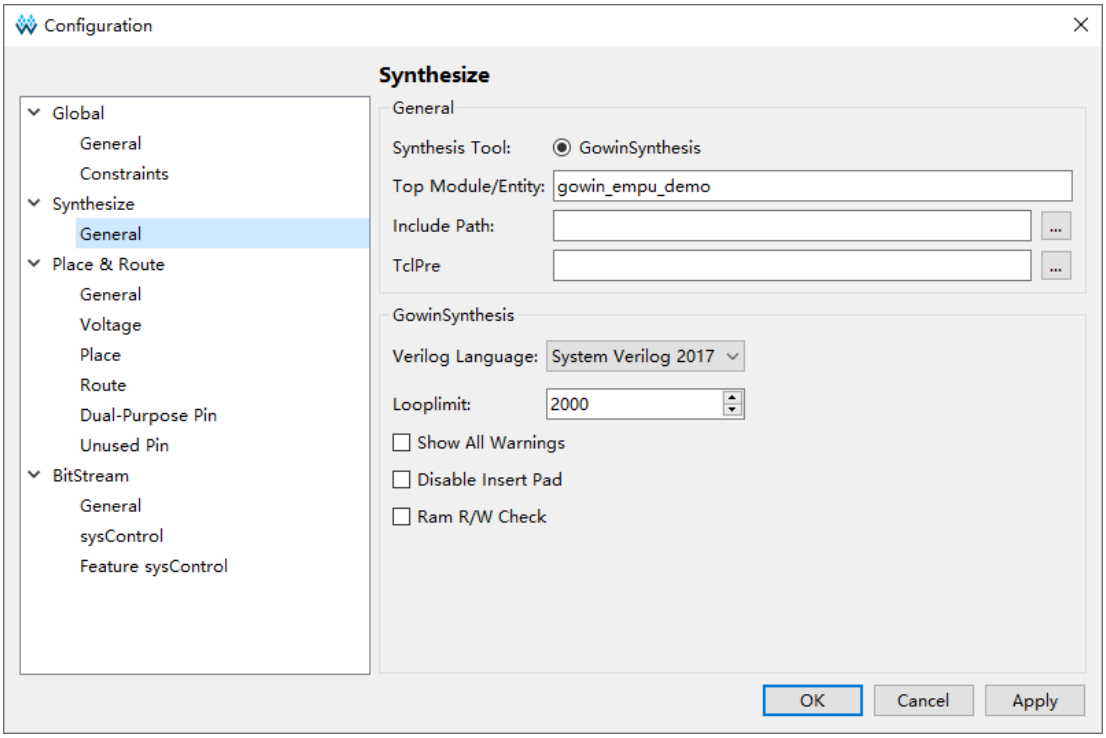
3.5 Configuration

3.5.1 Synthesize Configuration

The "Synthesize" configuration is as shown in Figure 3-17.

- Configure "Top Module/Entity" according to the top module name in the design.
- Configure "Include Path" according to the file path in the design.
- Configure Verilog Language, such as System Verilog 2017.

Figure 3-17 Synthesize Configuration

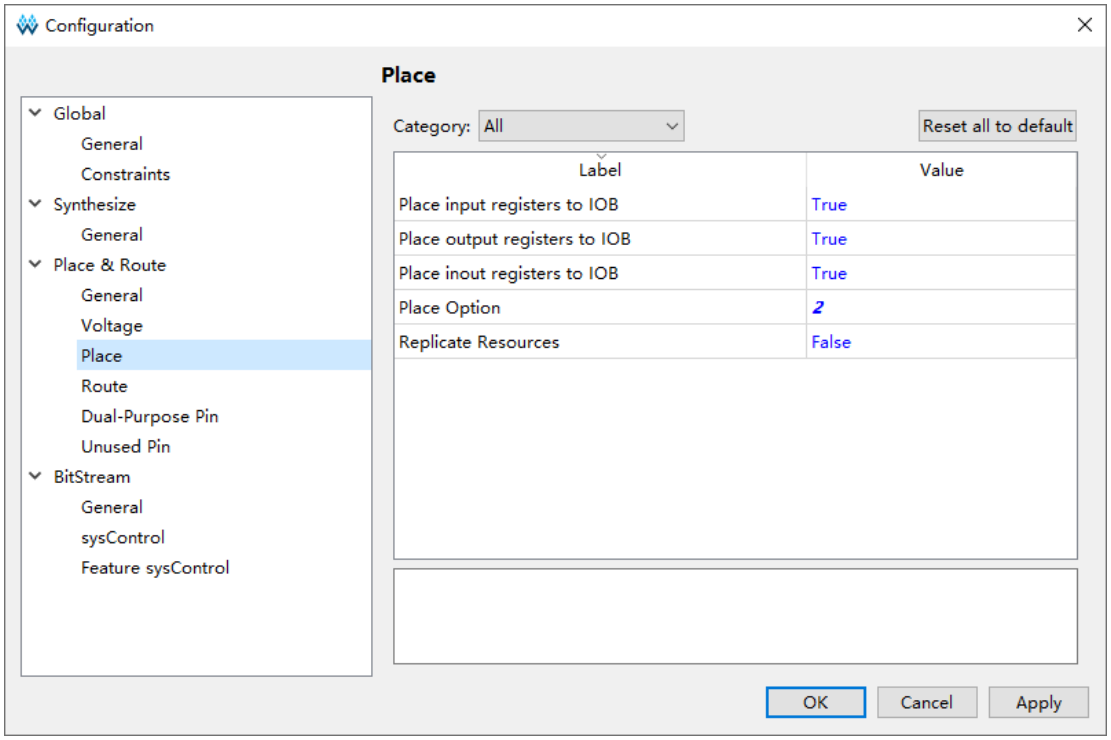


3.5.2 Place Configuration

Configure the "Place" option as shown in Figure 3-18.

For example, select "2" for "Place Option" to enhance the software timing.

Figure 3-18 Place Configuration

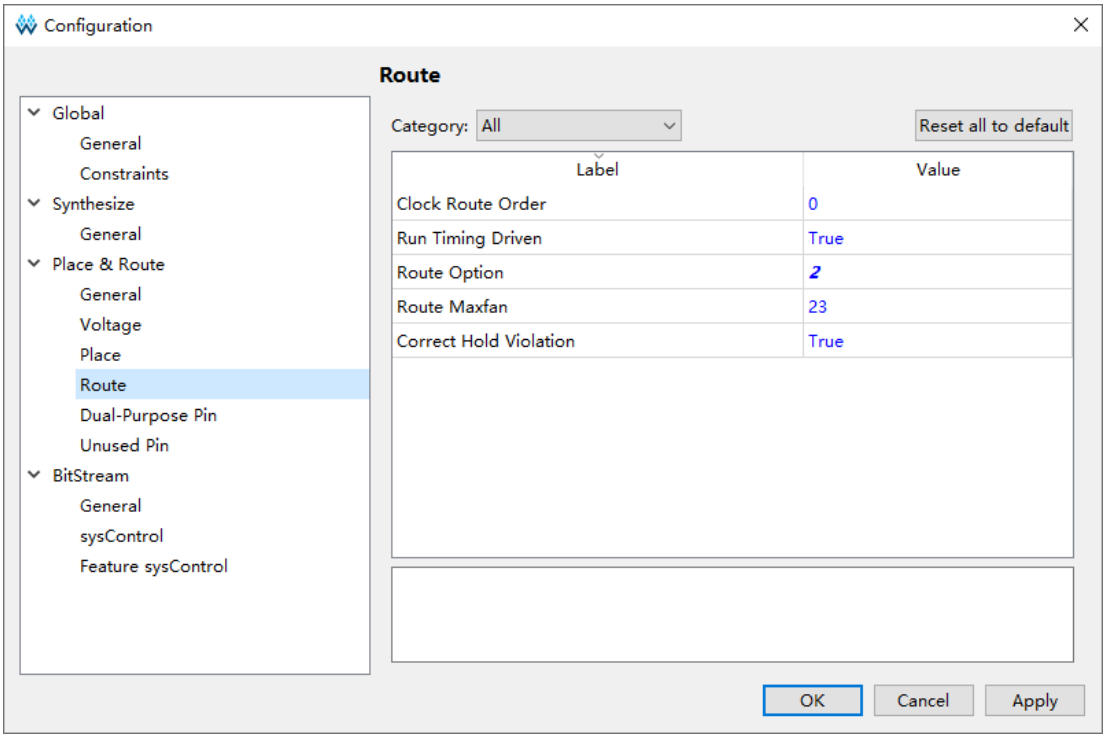


3.5.3 Route Configuration

Configure the "Route" option as shown in Figure 3-19.

For example, select "2" for "Route Option" to enhance the software timing.

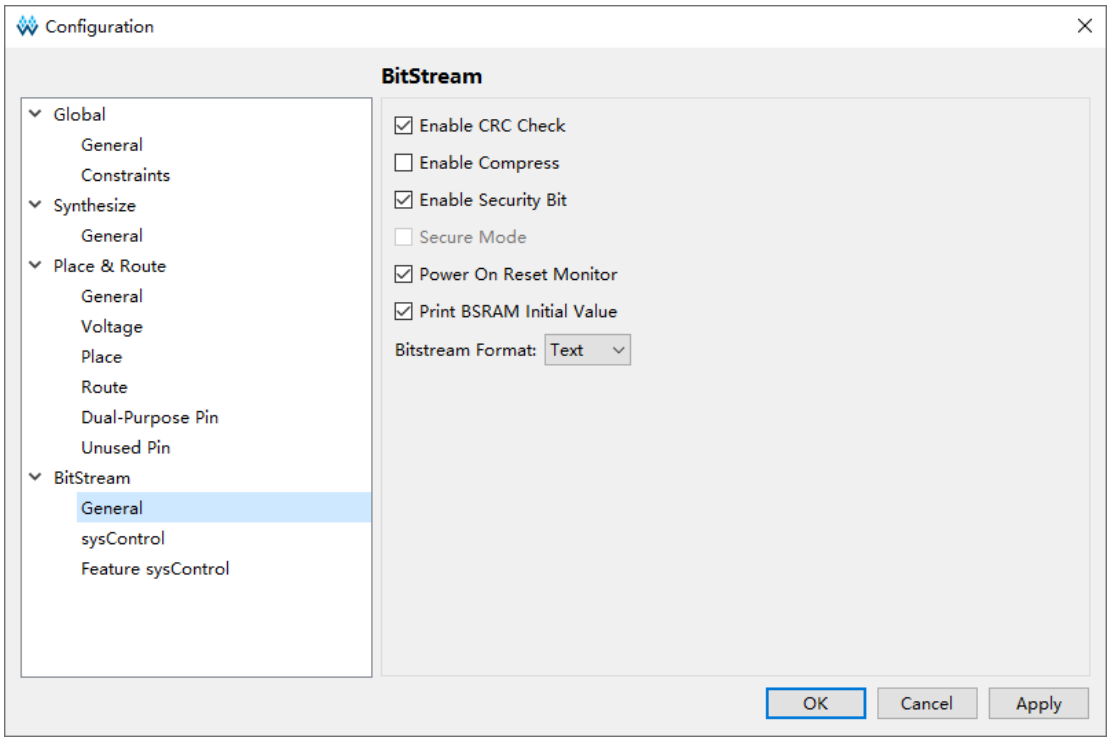
Figure 3-19 Route Configuration



3.5.4 Bitstream Configuration

Configure the "Bitstream" option as shown in Figure 3-20.

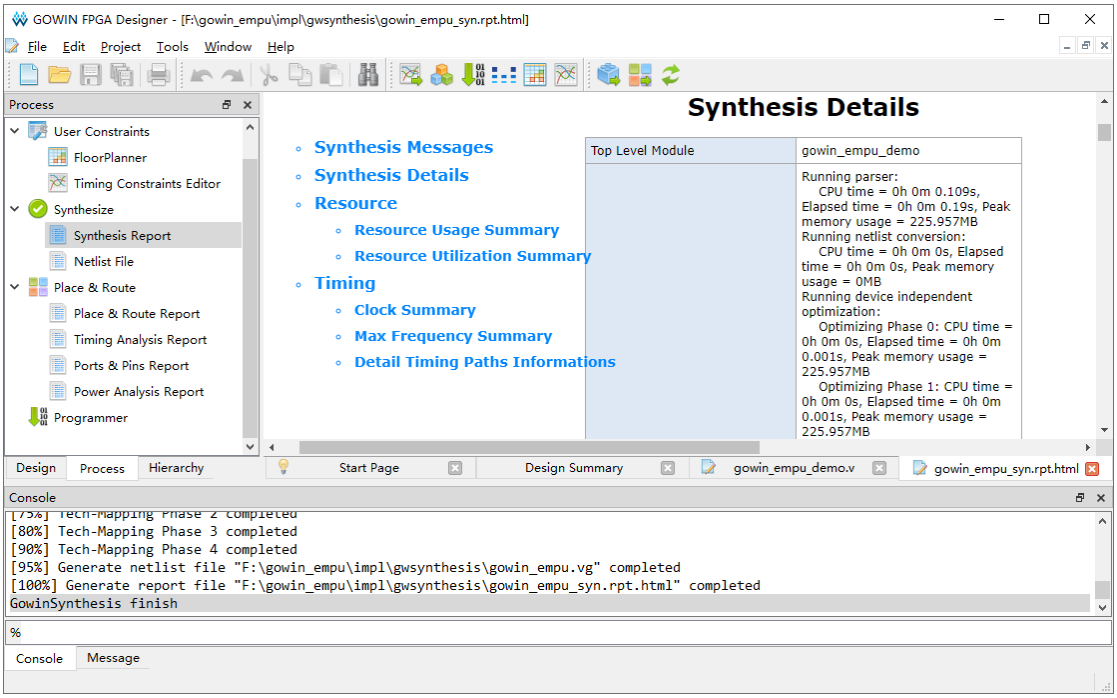
Figure 3-20 Bitstream Configuration



3.6 Synthesize

Run GowinSynthesis to complete the synthesis of RTL design to generate post-synthesis netlist file, as shown in Figure 3-21.

Figure 3-21 Synthesize

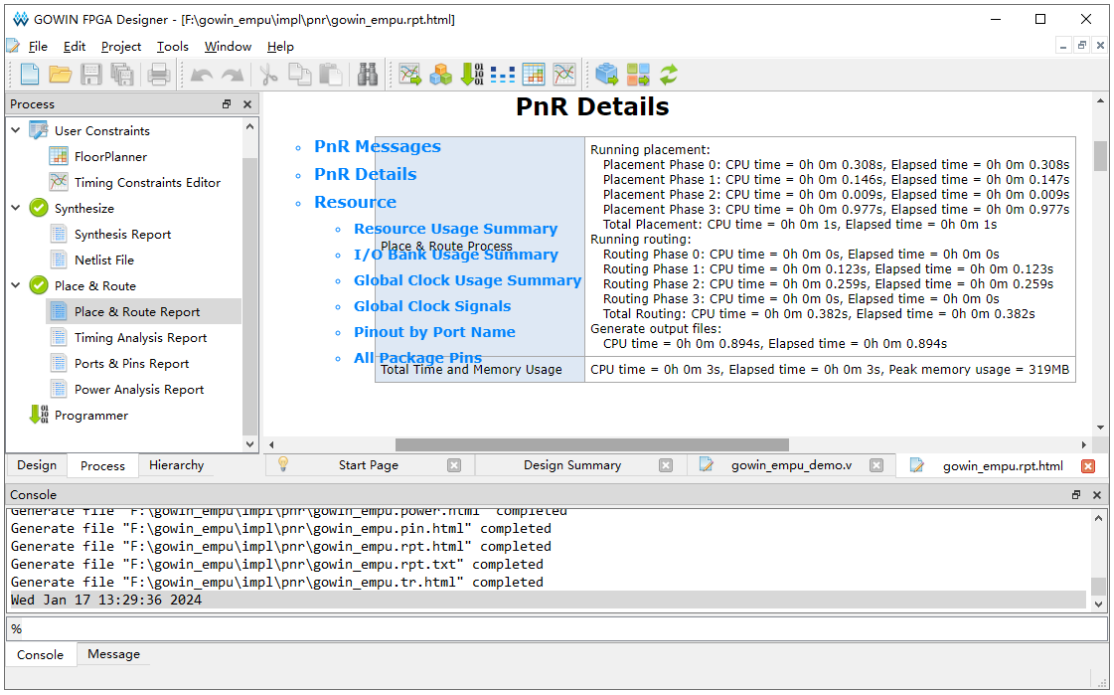


For the tool usage, please refer to *SUG100, Gowin Software User Guide*.

3.7 Place & Route

Run the Place & Route tool in Gowin software and generate the bitstream file, as shown in Figure 3-22.

Figure 3-22 Place & Route



For the tool usage, please refer to *SUG100, Gowin Software User Guide*.

3.8 Download

Download the bitstream file using Gowin Programmer.

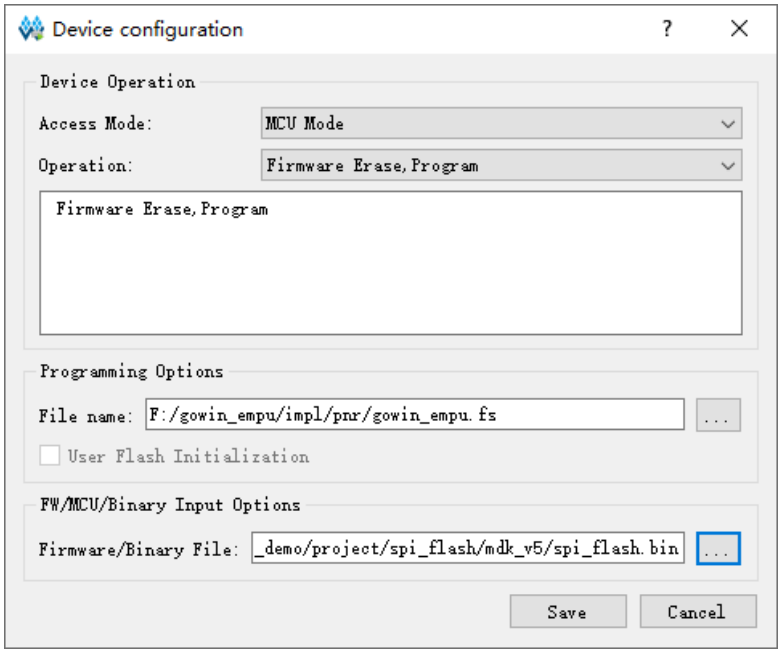
Open Programmer in Gowin software or under the installation path.

Click "Edit > Configure Device" on the menu bar or "Configure Device" ( ) on the tool bar to open the "Device configuration".

If FPGA product GW1NS-4C or GW1NSR-4C is selected, configuration options are as shown in Figure 3-23.

- Select "MCU Mode" in "Access Mode" drop-down list.
- Select "Firmware Erase, Program" or "Firmware Erase, Program, Verify" in "Operation" drop-down list.

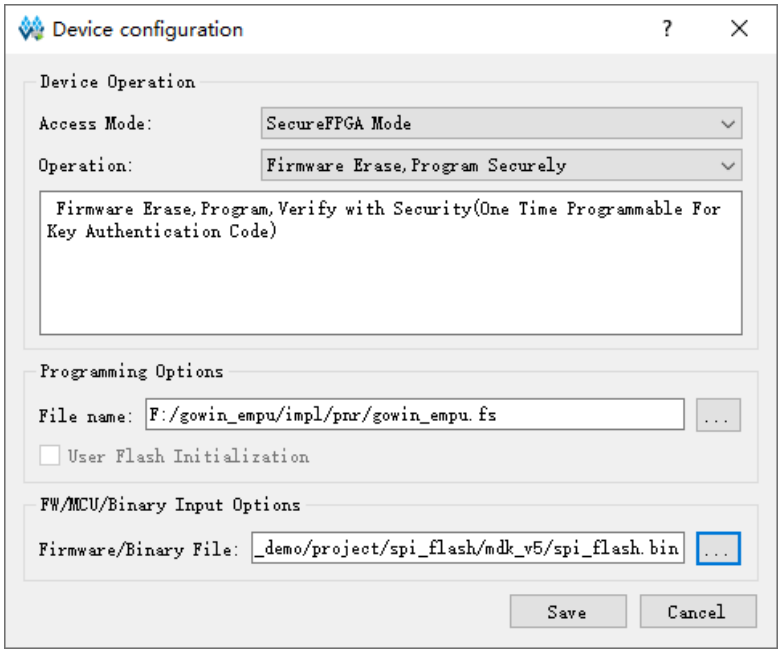
Figure 3-23 Download Configuration for GW1NS-4C/GW1NSR-4C



If FPGA product GW1NSER-4C is selected, configuration options are as shown in Figure 3-24.

- Select "SecureFPGA Mode" in "Access Mode" drop-down list.
- Select "Firmware Erase, Program Securely" in "Operation" drop-down list.

Figure 3-24 Download Configuration for GW1NSER-4C



- Import hardware design bitstream file in "Programming Options > File

name" .

- Click "Save" to complete the configuration.

Note!

Import software programming Binary file in " FW/MCU Input Options > Firmware/Binary File", please refer to *IPUG928, Gowin_EMPU(GW1NS-4C) IDE Software Reference Manual*.

After device configuration, click Program/Configure "  " on the Programmer tool bar to complete downloading of hardware design bitstream file and software programming Binary file.

For the usage of Programmer, please refer to *SUG502, Gowin Programmer User Guide*.

4 Reference Design

Gowin_EMPU(GW1NS-4C) provides hardware reference design in Gowin Software (tested software version V1.9.9.01 (64-bit)). You can click here to get the following reference design.

- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_MG64P_V1.1\gowin_empu\cm3_psram_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48G_V1.1\gowin_empu\cm3_i2c_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48G_V1.1\gowin_empu\cm3_spiflash_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48P_V1.1\gowin_empu\cm3_ahb2_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48P_V1.1\gowin_empu\cm3_apb2_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48P_V1.1\gowin_empu\cm3_demo
- …\ref_design\FPGA_RefDesign\DK_START_GW1NSR4C_QN48P_V1.1\gowin_empu\cm3_hyperram_demo
- …\solution\RTOS\ref_design\FPGA_RefDesign\gowin_empu
- …\solution\RunInSRAM_FromEmbFlash\ref_design\FPGA_RefDesign\gowin_empu
- …\solution\RunInSRAM_FromSIPFlash\ref_design\FPGA_RefDesign\gowin_empu


GOWIN
PROGRAMMING FOR THE FUTURE